

Práctica github Actions

Esta práctica consistirá en aplicar una serie de mejoras sobre un proyecto creado con el framework next.js ([link](#)) y que podéis descargar del siguiente repositorio de github ([link](#)). Para ello, partiendo de este código, deberéis crear un repositorio en vuestra cuenta de github y realizar las siguientes modificaciones:

Crear un workflow (nombre_repositorio_workflow) de github Actions con los siguientes jobs:

- (1 punto) Linter_job. Se encargará de ejecutar el linter que ya está instalado en el proyecto (existe un script para ello en el package.json) para verificar que la sintaxis utilizada es la correcta en todos los ficheros javascript. En caso de que existan errores deberéis corregirlos hasta que el job se ejecute sin problemas.
- (1,5 puntos) Cypress_job. Se encargará de ejecutar los tests de cypress ([link](#)) que contiene el proyecto. Para ello, utilizaréis la action oficial del proyecto ([link](#)). Si lo deseáis, podéis ejecutar manualmente mediante el comando npm run cypress (siempre que esté arrancado el proyecto previamente). Este job, que se ejecutará después del Linter_job, estará compuesto por los siguientes steps:
 - El encargado de realizar el checkout del código
 - El encargado de ejecutar los tests de cypress que continuará aunque se produzca un error (existe una propiedad que podéis establecer para conseguir este comportamiento)
 - El encargado de crear un artefacto (result.txt) que contendrá la salida del step anterior
- (2 puntos) Add_badge_job. Se encargará de publicar en el readme del proyecto el badge que indicará si se han superado los tests de cypress o no. Estará compuesto por los siguientes steps:
 - El encargado de realizar el checkout del código
 - El encargado de obtener los artefactos almacenados en el job anterior
 - Un step encargado de generar un output partiendo de la lectura del artefacto recuperado. Básicamente ejecutará la instrucción echo "::set-output name=cypress_outcome::\$(cat result.txt)"
 - Un step que ejecutará una action propia que deberéis crear. Esta action recibirá como parámetro de entrada el output generado por el step anterior y, dependiendo de si es "failure" o "success", modificará el README.md del proyecto añadiendo uno de los siguientes badges al final del mismo y tras el texto "RESULTADO DE LOS ÚLTIMOS TESTS":
 - (Failure) <https://img.shields.io/badge/test-failure-red>
 - (Success) <https://img.shields.io/badge/tested%20with-Cypress-04C38E.svg>
 - Step encargado de publicar el cambio del README.md en el repositorio
- (1,5 puntos). Deploy_job. Utilizando la action amondnet/vercel-action@v20, se encargará de publicar el proyecto en la plataforma vercel ([link](#)). Se ejecutará tras el Cypress_job y estará formado por dos steps:

- El encargado de realizar el checkout del código
- El encargado de desplegar la aplicación en vercel que utilizará la action `amondnet/vercel-action@v20`

PISTA: Para poder tener el proyecto funcionando correctamente en vercel os aconsejo revisar la documentación de la action ([link](#))

- (1,5 puntos) Job de envío de notificación a los usuarios del proyecto. Se denominará `Notification_job` y ejecutará, entre otros, una action propia que podrá basarse en Javascript. Se ejecutará siempre (aunque se haya producido un error en algún job previo), y se encargará de enviar un correo con:

Destinatario: dirección de correo vuestra personal tomada de un secret de github

Asunto: Resultado del workflow ejecutado

Cuerpo del mensaje:

Se ha realizado un push en la rama `main` que ha provocado la ejecución del workflow `nombre_repositorio_workflow` con los siguientes resultados:

- `linter_job`: resultado asociado
- `cypress_job`: resultado asociado
- `add_badge_job`: resultado asociado
- `deploy_job`: resultado asociado

PISTA: el estado final de los jobs ejecutados los podéis obtener a partir de variables de contexto que ofrece github ([link](#)).

- (1,5 punto) Configurad en vuestro readme personal (el del repositorio que tiene como nombre vuestro usuario de github) una action que permita mostrar métricas de los lenguajes más utilizados en los proyectos de vuestro perfil de github. Disponéis de la siguiente action ([link](#)). En este otro enlace ([link](#)) tenéis descritos los pasos que necesitáis realizar para poder configurar y usar la action correctamente. Tenéis total libertad de añadir el panel informativo de métricas que más os guste de entre todos los disponibles ([link](#)).

Una vez realizadas todas las acciones anteriores, documentad correctamente todos los pasos realizados en el `README.md` del proyecto (ojo!!!, es el mismo que modifica el job para añadir el badge así que id con cuidado). Por supuesto, deberá incluir una introducción teórica de los conceptos tratados (github-actions principalmente) (1 puntos)

Como respuesta a la actividad planteada, entregaréis el link a vuestro repositorio github así como el link de vercel donde estará desplegado el proyecto.